

# Exhibit D

RANDALL S. LUSKEY (SBN: 240915)

rluskey@paulweiss.com

**PAUL, WEISS, RIFKIND, WHARTON  
& GARRISON LLP**

535 Mission Street, 24th Floor

San Francisco, CA 94105

Telephone: (628) 432-5100

Facsimile: (628) 232-3101

ROBERT ATKINS (*Pro Hac Vice* admitted)

ratkins@paulweiss.com

CAITLIN E. GRUSAUSKAS (*Pro Hac Vice* admitted)

cgrusauskas@paulweiss.com

ANDREA M. KELLER (*Pro Hac Vice* admitted)

akeller@paulweiss.com

**PAUL, WEISS, RIFKIND, WHARTON  
& GARRISON LLP**

1285 Avenue of the Americas

New York, NY 10019

Telephone: (212) 373-3000

Facsimile: (212) 757-3990

*Attorneys for Defendants*

UBER TECHNOLOGIES, INC.,

RASIER, LLC, and RASIER-CA, LLC

*[Additional Counsel Listed on Following Page]*

**UNITED STATES DISTRICT COURT  
NORTHERN DISTRICT OF CALIFORNIA  
SAN FRANCISCO DIVISION**

IN RE: UBER TECHNOLOGIES,  
INC., PASSENGER SEXUAL  
ASSAULT LITIGATION

Case No. 3:23-md-03084-CRB

**DECLARATION OF WILLIAM ANDERSON  
IN SUPPORT OF DEFENDANTS UBER  
TECHNOLOGIES, INC., RASIER, LLC, AND  
RASIER-CA, LLC'S PROPOSED ESI  
PROTOCOL**

This Document Relates to:

ALL ACTIONS

Judge: Hon. Lisa J. Cisneros

Courtroom: G – 15th Floor

1 MICHAEL B. SHORTNACY (SBN: 277035)

2 mshortnacy@shb.com

3 **SHOOK, HARDY & BACON, L.L.P.**

4 2049 Century Park East, Suite 3000

5 Los Angeles, CA 90067

6 Telephone: (424) 285-8330

7 Facsimile: (424) 204-9093

8 PATRICK OOT (*Pro Hac Vice* admitted)

9 oot@shb.com

10 **SHOOK, HARDY & BACON, L.L.P.**

11 1800 K Street NW, Suite 1000

12 Washington, DC 20006

13 Telephone: (202) 783-8400

14 Facsimile: (202) 783-4211

15 KYLE N. SMITH (*Pro Hac Vice* admitted)

16 ksmith@paulweiss.com

17 JESSICA E. PHILLIPS (*Pro Hac Vice* admitted)

18 jphillips@paulweiss.com

19 **PAUL, WEISS, RIFKIND, WHARTON**  
20 **& GARRISON LLP**

21 2001 K Street, NW

22 Washington DC, 20006

23 Telephone: (202) 223-7300

24 Facsimile: (202) 223-7420

25 *Attorneys for Defendants*

26 UBER TECHNOLOGIES, INC.,

27 RASIER, LLC, and RASIER-CA, LLC

1 I, William Anderson, hereby declare and state:

2 1. I am a Manager, eDiscovery at Uber Technologies Inc. (“Uber”). I have been employed at  
3 Uber since October 2015, and I have held my current position since March 2024. I previously held  
4 the positions of Senior Discovery Analyst II, Senior eDiscovery Analyst, Executive Support  
5 Technician, and Computer Technician at Uber. Over the course of my employment at Uber, I have  
6 acquired personal knowledge of Uber’s discovery capabilities concerning its communication tools  
7 and data sources. I also note that Uber is the 2024 recipient of Legalweek’s award for E-Discovery  
8 and Litigation Technology, In-House.

9 2. The contents of this declaration are true and correct to the best of my knowledge,  
10 information, and belief, and are based on my personal knowledge of Uber’s discovery capabilities as  
11 they relate to Uber’s communication tools and data sources and/or informed conversations with  
12 knowledgeable employees at Google and Lighthouse (Uber’s eDiscovery vendor). If called upon as a  
13 witness in this action, I could and would testify competently to the matters discussed in this  
14 declaration.

15 3. I have worked with Google Workspace (formerly G Suite) since I started my employment at  
16 Uber eight-and-a-half years ago. I have significant experience addressing the challenges of  
17 collecting Google Drive files referenced in a URL, including collecting specific “versions” of  
18 Google Drive files.

19 4. Over the past several years, Uber has gone through a myriad of steps to develop a process to  
20 collect hyperlinked Google Drive documents for production. I worked with Uber’s eDiscovery  
21 vendor Lighthouse to develop their proprietary “Google Parser,” which is not a document collection  
22 tool but is useful in identifying linked documents within collected documents. That development and  
23 implementation process took well over a year to complete.

24 5. I have also devoted significant time and resources to investigate whether there is a tool or a  
25 method to automate the trial-and-error, manual process of identifying and collecting a particular  
26 “version” of such a Google Drive file. In addition to all of this past work, and with respect to the  
27 issue presented in this litigation, I have spoken numerous times with Uber’s eDiscovery vendor  
28

1 Lighthouse; an Enterprise Google Workspace Specialist, Cloud Customer Engineering at Google; a  
 2 Technical Account Manager at Google; and a Google Collaboration and Productivity Specialist at  
 3 Google who all have acknowledged that such collection is a manual, one-URL-at-a-time process;  
 4 that they are not aware of a program or tool that automates the process; and that they do not believe  
 5 that such automation is feasible based on the current Google Workspace environment.

### 6 **The Technical Challenges of Hyperlinks**

7 6. Hyperlinked documents are not attachments. Hyperlinked documents present unique  
 8 collection, review, and production challenges that make them different from the standard collection,  
 9 review, and production of attachments to emails and other communications.

10 7. Uber has used or currently uses several different collaboration and communication tools that  
 11 an employee may use to share or generate a hyperlink, including Google Mail, Google Chat, Slack,  
 12 uChat, HipChat, Box, among others. An Uber employee can create or provide a link to any of these  
 13 systems that links to a document in one of the other, unrelated systems.

14 8. In my experience, and to the best of my knowledge, there is no discovery tool or application  
 15 (neither within Uber's technology, and nor within the commercial marketplace generally) that  
 16 automatically identifies, collects, and associates underlying hyperlinked documents to documents or  
 17 email messages containing embedded hyperlinks to various, unrelated systems. Searching for,  
 18 collecting, and linking underlying hyperlinked documents to documents containing embedded  
 19 hyperlinks to unrelated systems must be done manually, on a source-by-source, hyperlink-by-  
 20 hyperlink basis. Further, each source system presents its own unique challenges when it comes to  
 21 manually identifying and collecting hyperlinked documents.

### 22 **The Specific Challenges of Collecting a Particular "Version" of a Google Drive File from a** 23 **URL that Appears within a Gmail Message**

24 9. Even when Uber works with eDiscovery vendors to produce documents, Uber's eDiscovery  
 25 team performs its own document collections, which it then exports to its vendors. Uber's vendors  
 26 and other third parties do not have access to Uber's information systems.

1 10. Google Workspace is a leading business solution used by Uber and countless other  
2 sophisticated companies, including other Fortune 500 companies. Google Workspace offers a suite  
3 of cloud-based web applications and file storage solutions, such as Gmail, Chat, Drive, Docs, and  
4 Sheets. Google Workspace Enterprise edition also offers Google Vault. To access files within  
5 Google Workspace, there are different entry points: Google user interface, Google Drive API, and  
6 Google Vault API.

7 11. Google user interface is what the user sees when accessing, for instance, Google Drive or  
8 Google Vault.

9 12. API stands for Application Program Interface, which is a way to interact with a program  
10 outside of the regular user interface.

11 13. Google Drive is a system that allows users to store files in the cloud, synchronize files across  
12 devices, and share files.

13 14. Google Vault is an information governance and eDiscovery tool for Google Workspace.  
14 Uber uses Google Vault to retain and hold users' Google Workspace data, which can include, among  
15 other things, Gmail messages and Google Drive files. Because Google Vault retains and holds users'  
16 Google Workspace data for information governance and eDiscovery purposes, Google Vault is the  
17 appropriate source from which to collect documents.

18 15. For this reason and others, Uber's standard discovery process involves, for example,  
19 exporting a custodian's Gmail messages and Google Drive files from Google Vault. The export  
20 provides metadata that Uber's eDiscovery vendor can then include within document productions,  
21 which allows a receiving party to create an association between a Gmail message and a Google  
22 Drive file linked in the Gmail message.

23 16. My understanding is that where a given Gmail message contains a link to a Google Drive  
24 document, Plaintiffs want Uber to produce the Gmail message along with the contemporaneous  
25 "version" of that document in the form it existed at the time it was sent by the email user and to  
26 produce the Gmail message together with that "version" of that document in a "family" relationship.  
27  
28

1 But there is no automatic process for collecting specific point-in-time “versions” of Google Drive  
2 documents.

3 17. Rather, in my experience, the only way to collect a specific point-in-time “version” of a  
4 Google Drive file from a URL that appears within a Gmail message is to undertake a one-URL-at-a-  
5 time manual process. To reiterate, in my experience, and to the best of my knowledge, no tool  
6 (within Uber or in the general commercial marketplace) exists that allows for automatic, non-manual  
7 collection of specific “versions” of Google Drive files that are referenced by a URL. In particular,  
8 Google itself has not developed such a process. In my experience, including through one-at-a-time  
9 collections in limited circumstances, an automatic process cannot be developed because there are too  
10 many failure points that occur when collecting a Google Drive file that is referenced by a URL.

11 18. For example, a user first would need to create an API project within Google Workspace. An  
12 API project is a collection of settings and credentials that allows an individual user to make use of  
13 Google APIs.

14 19. To begin the process of collecting a point-in-time “version” of a particular Google Drive file,  
15 Uber first would need to identify a Gmail message that contains a URL referencing that Google  
16 Drive file.

17 20. The administrator credentials (that would be required to execute this step) cannot access  
18 documents without assuming the identity of a specific email user who either owns or has proper  
19 access permission to the Google Drive file.

20 21. Thus, using administrator credentials for the hypothetical API project such as being proposed  
21 here, a user would attempt to use the email address of the sender of the Gmail message and the  
22 Google Document ID for the URL-referenced document to run a query through the API project for  
23 the revision history. The revision history is a list of “versions” of the document created by Google,  
24 and the revision history includes the revision numbers for the various “versions.”

25 22. But an issue that often occurs is that the email address used for that query does not have  
26 access to the Google Drive file, and therefore that query will fail. Just because a Google Drive file is  
27  
28

1 linked in a Gmail message does not mean that everyone, or anyone, on that Gmail message chain has  
2 appropriate access to obtain the revision history for that Google Drive file.

3 23. In the event that the Gmail sender does not have appropriate access to the Google Drive  
4 document, the user would then search the Gmail message for the email addresses of recipients.

5 24. The user would then have to attempt new queries, *one at a time*, for each recipient email  
6 address, to determine whether any of those email recipients had proper access permission to the  
7 Google Drive file in order to obtain the revision history.

8 25. But there are instances where those additional queries fail to pull the revision history because  
9 the Gmail message recipients do not have proper access to the Google Drive file.

10 26. When that occurs, the user would need to undertake additional research in an effort to  
11 identify an email address that has appropriate access to the Google Drive file in order to even obtain  
12 or gain access to the revision history.

13 27. Assuming the user is ultimately successful in pulling the revision history, based on  
14 experience, the initial query for the revision history regularly outputs a non-exhaustive revision  
15 history.

16 28. When that occurs, the user would then need to manually intervene by creating additional  
17 parameters for new queries until the user, based on a review of all revision histories pulled, can  
18 confirm that all available “versions” of the Google Drive file have been identified.

19 29. Once the user has a complete revision history for a particular Google Drive file, the user  
20 would need to search that revision history to identify the “version” with a time stamp that is closest  
21 to, but not later than, the time stamp of the Gmail message. The time zone in the Gmail message  
22 does not necessarily align with the time zone in the revision history, requiring yet another manual  
23 step whereby the user must convert the time zone in the Gmail message to align with the time zone  
24 in the revision history.

25 30. From there, the user would need to start an entirely new process to export that identified  
26 “version” of the Google Drive file. Uber is able to search for and export Google Drive files from  
27 Google Drive and Google Vault. To export a version from a specific time in the day, the export  
28

1 would have to take place from Google Drive because Google Vault allows exports only by  
2 specifying a date without any additional granularity.

3 31. To attempt an export of an identified “version” from Google Drive, Uber would need to input  
4 in the API project the Google Document ID, the email address of the owner or someone with proper  
5 access permission to the document, and the revision number, as identified from the revision history.

6 32. If that export is successful, the user would need to manually rename the file with the revision  
7 number as a suffix, to avoid file-name duplication.

8 33. The user would then need to run a separate query in the API project using the Google  
9 Document ID to locate the metadata for that document.

10 34. The user would then need to use the identified metadata to create a metadata file for that  
11 particular “version” of the Google Drive file. To be clear, the metadata for any particular Google  
12 Drive file will be the same regardless of the “version.” There is only one set of metadata for a  
13 Google Drive file, and it is the metadata as reflected as of the date of collection.

14 35. But often, based on my experience, an export out of Google Drive does not work (e.g., in  
15 instances where the Google Drive file exists only within Google Vault).

16 36. If the export from Google Drive does not work, the user would need to start a new process to  
17 attempt to export the document from Google Vault.

18 37. In that new process, the user would need to search the Google Vault API by using the title of  
19 the document as it appears in the Gmail message, the email address of the owner or someone with  
20 proper access permission, and revision date.

21 38. If the Gmail message does not contain the title of the document, the user would need to  
22 conduct a separate search to try to find the document manually.

23 39. Notably, the user cannot search the Google Vault API by using the Google Document ID.

24 40. If the title of the document has changed since the Gmail message was sent, the search likely  
25 will not work.

1 41. Even if the search of the Google Vault API is successful, the search will often return multiple  
2 documents that have the same or similar titles, requiring the user to manually review the returned  
3 documents to identify the correct one.

4 42. If that export is successful, the user would need to rename the file with the revision number  
5 as a suffix, to avoid file-name duplication.

6 43. To import the document to a review platform, for each link, the eDiscovery vendor would  
7 need to determine how to add identifying information, such as the document control number, to that  
8 specific Gmail message and URL. Without this information, the relationship between the Google  
9 Drive file “version” and the specific Gmail message and URL will not be tracked.

10 44. The eDiscovery vendor would further need to implement a way to avoid de-duplication of the  
11 various “versions” of the same Google Drive file.

12 45. This above-described process is the process that would need to be followed to retrieve a  
13 single “version” of a single document. This manual process would not scale to the collection of  
14 thousands or millions of Google Drive file “versions,” which would be infeasible.

15 46. Based on the required manual intervention throughout this process described above, Google  
16 Vault’s API or a macro recorder (which would simply record keystrokes on a computer) could not be  
17 used to automate the process.

18 **Plaintiffs Propose that Uber Attempt to Invent a Solution that Does Not Exist and Would Not**  
19 **Work Here to Automate the Process of Collecting Specific “Versions” of Google Drive Files**  
20 **Referenced by URL in Gmail Messages**

21 47. I have reviewed “Plaintiffs Proposed Methodology for Retrieving Google Drive files Linked  
22 to Within Gmails” (“Plaintiffs’ Proposal”).

23 48. Plaintiffs’ Proposal does not identify an existing tool, but rather suggests that Uber attempt to  
24 invent a tool that no other company, not even Google, has created to date and that will not work.

25 49. Notably, Plaintiffs’ Proposal does not recommend that Uber use Metasploit Forensic  
26 Evidence Collector or a macro recorder.

1 50. Plaintiffs' Proposal also does not address collection of Google Drive files in sources other  
2 than Gmail messages.

3 51. Plaintiffs' Proposal is that a "computer programmer" will attempt to create a program that  
4 can somehow overcome all of the challenges and failure points that, as described above, occur when  
5 attempting to collect a particular "version" of a Google Drive file referenced in a Gmail message,  
6 and that require manual intervention. But no such programming solution is feasible here as  
7 evidenced by the fact that, to my knowledge, not a single company offers a product that automates  
8 this work, not even Google. Further, Plaintiffs' Proposal describes the proposed program in just five  
9 sentences. *See* Pls.' Proposal at ¶ 5.a.–c. Again, no program exists that performs these cycling,  
10 retrieving, recording, comparing, converting, and downloading functions that Plaintiffs' Proposal  
11 vaguely describes, and also there is no evidence that such a program could be created or would  
12 work. Aside from the fact that Uber does not employ a computer programmer with the unique skill  
13 set for this specific proposal Plaintiffs put forward, even if there were such a tool (and there is not),  
14 to test, develop, and implement it within the computer environment of a corporation could take at  
15 least a year to complete.

16 52. For example, Plaintiffs' Proposal states the programmer will create a program that "Cycles  
17 through the Missing List and for each link extracts the Google Document ID and then retrieves its  
18 revision list, recording whether the retrieval was successful or failed, and if failed, any reported error  
19 code(s)." *See* Pls.' Proposal at ¶ 5.a.

20 53. Not only does this sentence skip necessary steps outlined above, but multiple failure points  
21 exist within that single sentence.

22 54. As described above, a user can access a Google Drive file's revision list only with the email  
23 address of someone who owns or has appropriate access permission to the Google Drive file. Often,  
24 the sender and recipient of the Gmail message containing the URL reference do not have appropriate  
25 access permission to the Google Drive file. In those instances, the user needs to instigate an  
26 investigation to determine whether the user can identify through other means an email address of a  
27 person with appropriate access permission.  
28

1 55. Also, as described above, retrieval of the revision list is often not accomplished through a  
2 single query. Rather, manual intervention is often required to formulate multiple additional queries  
3 until a complete revision history is obtained.

4 56. As another example, Plaintiffs' Proposal states, "If the correct revision is located, convert the  
5 file to its Microsoft equivalent and download it along with appropriate metadata." *See* Pls.' Proposal  
6 at ¶ 5.c.

7 57. But this incorrectly assumes that Google Drive files can be accessed from the revision list.  
8 They cannot. Rather, the user would need to attempt to locate and export the document in  
9 accordance with the processes described above. Those processes include queries for attempting to  
10 export a document and its metadata, which are separate queries, from Google Drive, as well as yet  
11 another a separate process requiring manual intervention to export a document from Google Vault.  
12 Further, not all file types stored in Google Drive have a Microsoft equivalent. There are also the  
13 separate required processes of creating metadata files and renaming the documents to avoid file-  
14 name collisions. None of which is accounted for within Plaintiffs' single sentence. Again, I am not  
15 aware of any program in existence that can perform these tasks, nor would Plaintiffs' proposed  
16 program accomplish these tasks.

17 58. Plaintiffs' Proposal states in a footnote that "[t]he program steps outlined here can all be  
18 accomplished used [sic] the approach set out in" and then provides the following URL link to a  
19 stackoverflow.com web forum: [https://stackoverflow.com/questions/77467875/revert-to-specific-](https://stackoverflow.com/questions/77467875/revert-to-specific-version-of-google-sheets-with-respect-to-specific-date-using#:~:text=This%20sample%20script%20is%20for%20Drive%20API%20v3)  
20 [version-of-google-sheets-with-respect-to-specific-date-](https://stackoverflow.com/questions/77467875/revert-to-specific-version-of-google-sheets-with-respect-to-specific-date-using#:~:text=This%20sample%20script%20is%20for%20Drive%20API%20v3)  
21 [using#:~:text=This%20sample%20script%20is%20for%20Drive%20API%20v3](https://stackoverflow.com/questions/77467875/revert-to-specific-version-of-google-sheets-with-respect-to-specific-date-using#:~:text=This%20sample%20script%20is%20for%20Drive%20API%20v3) ("Stack Overflow  
22 Q&A"). Stack Overflow is a question-and-answer website where third-party individuals using the  
23 website may ask and answer computer-programming questions.

24 59. First, as outlined above, "[t]he program steps [Plaintiffs] outlined" in paragraph 5 of  
25 Plaintiffs' Proposal are insufficient to accomplish the tasks of identifying and collecting specific  
26 "versions" of Google Drive files referenced by URL in Gmail messages.

1 60. Second, even “[t]he program steps [Plaintiffs] outlined” in Paragraph 5 of Plaintiffs’  
2 Proposal are not covered by the scripts listed in the Stack Overflow Q&A, which do not address  
3 cycling through URLs to extract Google Document IDs, retrieving revision lists, comparing listed  
4 revisions to Gmail messages, and recording outcomes.

5 61. In the linked Stack Overflow Q&A, an individual going by the handle “EagleEye” asked how  
6 to “get the version of my Google sheet dated back to 31st December 2022.” In response, an  
7 individual going by the handle “Tanaike” first confirmed that “EagleEye” was not looking for a  
8 particular “version,” but rather for any “version” of the document from that date. “Tanaike” then  
9 provided two sample scripts, one for Drive API v2 and one for Drive API v3. But “EagleEye”  
10 responded that the script did not work because, regardless of which revision date “EagleEye” input,  
11 there was only one output: “No revisions in \${date}.” “Tanaike” responded, “I apologize for the  
12 inconvenience. Unfortunately, I cannot replicate your situation. I think that this is due to my very  
13 poor skill. But, from your reply, I’m worried that the revision data might not be able to be correctly  
14 retrieved.” While the script allegedly worked for “Tanaike,” he could not explain why the script was  
15 not working for “EagleEye.” In further response, he said, “I deeply apologize for this situation. I  
16 cannot replicate your situation.” “EagleEye” did not reply, and that was the end of the exchange.

17 62. As an initial matter, according to the anonymous internet user “EagleEye,” this proposed  
18 script did not work. But even a functioning version of this script would not address the issues  
19 presented here. This script was designed for a single document using the Google Drive API,  
20 restoring a non-Vault document, with owner access. This script would not work for Google Vault.  
21 Additionally, the script assumes that the user has already identified the “version” date and Document  
22 ID, and then requires the user to manually input that information into the script. Further, the script  
23 does not address the issue of access permissions because “EagleEye” was searching for his or her  
24 own document. The script also provides guidance for locating only the latest or earliest “version” on  
25 a particular date, not the “version” contemporaneous to a particular Gmail message.

**Metaspike's Forensic Evidence Collector Would Not Work Here to Automate the Process of Collecting Specific "Versions" of Google Drive Files Referenced by URL in Gmail Messages**

63. Uber does not utilize Metaspike's Forensic Email Collector ("FEC") because, among other reasons, it would not provide Uber any benefit in its discovery processes, and therefore there is no reason to undertake the expense and burden of purchasing, installing, integrating, testing, and maintaining this unfamiliar tool within Uber's environment.

64. FEC cannot access items stored in Google Vault. Uber would first need to export the Google Workspace data from Google Vault, which would defeat the purpose because, as described above, Google Vault exports only the current "version" of Google Drive files.

65. Further complicating the matter, FEC is a desktop application, which slows the collection process. FEC also has known and disclosed throttling issues whereby the number of requests exceed Google's permissible limits, slowing down the collection process and creating collection errors. *See* <https://docs.metaspike.com/article/42-throttling-definition-workaround>. These known issues would create yet further inefficiencies, if not complete obstacles, to collecting Google Drive data.

66. FEC also cannot identify a hyperlink in one platform (for example, in Slack) and then identify, collect, and associate the hyperlinked document if the hyperlinked document exists in a separate, unrelated platform (for example, in Google Vault). Further, FEC supports hyperlink association and collection only within the Google Workspace platform. Accordingly, FEC would not be capable of collecting any hyperlinks in communications or documents located in other, non-Google systems that are not compatible with FEC, such as Slack, uChat, and HipChat.

67. FEC would therefore not resolve the obstacles identified above if applied to Uber's data environment or systems, and would increase the amount of time and resources that Uber would need to devote to its discovery processes.

**Conclusion**

68. To my knowledge, no tool exists, or is feasible to create, that allows for automatic collection of specific "versions" of Google Drive files from a URL.

1 69. FEC, which Plaintiffs' Proposal does not reference or suggest Uber should use, also would  
2 not work.

3 70. For these reasons, Plaintiffs' Proposal imposes a substantial burden on Uber while still not  
4 providing the ability to collect hyperlinked documents without a case-by-case, hyperlink-by-  
5 hyperlink manual process.

6 I declare under penalty of perjury that the foregoing is true and correct, and that I executed this  
7 Declaration on April 12, 2024, in Denver, CO.

8 *William Anderson*

9 \_\_\_\_\_  
10 William Anderson  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28